CLAIMS

What is claimed is:

1. A method comprising:

identifying data to be signed;

establishing parameter data for use with signature generating logic that encrypts data based on a Jacobian of a curve, said Jacobian having a genus exceeding one, said parameter data causing said signature generating logic to select at least one Gap Diffie-Hellman (GDH) group of elements relating to said curve;

determining private key data and corresponding public key data using said signature generating logic; and

signing said identified data with said private key data using said signature generating logic to create a corresponding digital signature.

2. The method as recited in Claim 1, wherein said identified data includes a message $m \in \{0, 1\}^*$.

3. The method as recited in Claim 2, wherein said parameter data establishes a base group G and a generator g as system parameters for said signature generating logic.

4. The method as recited in Claim 3, wherein determining said private key data and said public key data includes:

picking $x \overset{R}{\leftarrow} Z_p^*$; and

computing $v \leftarrow g^x$, wherein said public key data includes v and said private key data includes x.

5. The method as recited in Claim 4, wherein signing said identified data using with said private key data using said signature generating logic further includes:

determining $h \leftarrow h(m)$, and $\sigma \leftarrow h^x$, using at least one hash function, said private key data x and said message m, wherein said digital signature includes σ.

6. The method as recited in Claim 5, wherein said hash function includes a full-domain hash function h: $\{0, 1\}^* \rightarrow G$.

7. The method as recited in Claim 5, wherein said hash function includes a hash function h':$\{0, 1\}^* \rightarrow G \cup \{\perp\}$, that outputs an element of G or $\perp$ indicating a failure.

8. The method as recited in Claim 1, further comprising:

outputting said digital signature.

9. The method as recited in Claim 8, further comprising:

determining if said digital signature is valid using signature verifying logic.

10. The method as recited in Claim 9, wherein said signature verifying logic is configured using said parameter data and said parameter data establishes a

base group G and a generator g as system parameters for said signature verifying logic.

11. The method as recited in Claim 10, wherein:

said public key data includes public key data v;

said identified data includes a message m;

said digital signature includes signature σ; and

determining if said digital signature is valid using said signature verifying logic further includes:

determining $h \leftarrow h(m)$ using at least one hash function, and

verifying that (g, v, h, σ) is a valid Gap Diffie-Hellman tuple.

12. The method as recited in Claim 1, wherein said digital signature is included in a product ID.

13. A computer-readable medium having computer implementable instructions for causing at least one processing unit to perform acts comprising:

providing signature generating logic capable of digitally signing identified data;

configuring said signature generating logic using parameter data, said signature generating logic being configured to digitally sign said identified data based on a Jacobian of a curve, said Jacobian having a genus greater than one, said parameter data causing said signature generating logic to select at least one Gap Diffie-Hellman (GDH) group of elements relating to said curve;

determining private key data and corresponding public key data using said signature generating logic; and

signing said identified data with said private key data using said signature generating logic to create a corresponding digital signature.

14. The computer-readable medium as recited in Claim 13, wherein said identified data includes a message m $\in$ $\{0, 1\}^*$.

15. The computer-readable medium as recited in Claim 14, wherein said parameter data establishes a base group G and a generator g as system parameters for said signature generating logic.

16. The computer-readable medium as recited in Claim 15, wherein determining said private key data and said public key data includes:

picking $x \overset{R}{\leftarrow} Z_p^*$; and

computing $v \leftarrow g^x$, wherein said public key data includes v and said private key data includes x.

17. The computer-readable medium as recited in Claim 16, wherein signing said identified data using with said private key data using said signature generating logic further includes:

determining $h \leftarrow h(m)$, and $\sigma \leftarrow h^x$, using at least one hash function, said private key data x and said message m, wherein said digital signature includes σ.

18. The computer-readable medium as recited in Claim 17, wherein said hash function includes a full-domain hash function h: $\{0, 1\}^* \rightarrow G$.

19. The computer-readable medium as recited in Claim 17, wherein said hash function includes a hash function h':$\{0, 1\}^* \rightarrow G \cup \{\perp\}$, that outputs an element of G or $\perp$ indicating a failure.

20. The computer-readable medium as recited in Claim 13, further comprising:

outputting said digital signature.

21. The computer-readable medium as recited in Claim 20, further comprising:

determining if said digital signature is valid using signature verifying logic.

22. The computer-readable medium as recited in Claim 21, wherein said signature verifying logic is configured using said parameter data and said parameter data establishes a base group G and a generator g as system parameters for said signature verifying logic.

23. The computer-readable medium as recited in Claim 22, wherein:

said public key data includes public key data v;

said identified data includes a message m;

said digital signature includes signature σ; and

determining if said digital signature is valid using said signature verifying logic further includes:

determining $h \leftarrow h(m)$ using at least one hash function, and

verifying that (g, v, h, σ) is a valid Gap Diffie-Hellman tuple.

24. An apparatus comprising:

memory configured to store identifying data that is to be signed;

signature generating logic that encrypts data based on a Jacobian of a curve, said Jacobian having a genus greater than one, said signature generating logic being operatively coupled to said memory and configurable using parameter data, said parameter data causing said signature generating logic to select at least one Gap Diffie-Hellman (GDH) group of elements relating to said curve, and wherein said signature generating logic determines private key data and corresponding public key data, and then signs said identified data with said private key data to create a corresponding digital signature.

25. The apparatus as recited in Claim 24, wherein said identified data includes a message $m \in \{0, 1\}^*$.

26. The apparatus as recited in Claim 25, wherein said parameter data establishes a base group G and a generator g as system parameters for said signature generating logic.

27. The apparatus as recited in Claim 26, wherein said signature generating logic determines said private key data and said public key data by:

picking $x \in_R Z_p^*$; and

computing $v \leftarrow g^x$, wherein said public key data includes v and said private key data includes x.

28. The apparatus as recited in Claim 27, wherein said signature generating logic is further configured to:

determine $h \leftarrow h(m)$, and $\sigma \leftarrow h^x$, using at least one hash function, said private key data x and said message m, wherein said digital signature includes σ.

29. The apparatus as recited in Claim 28, wherein said hash function includes a full-domain hash function h: $\{0, 1\}^* \rightarrow G$.

30. The apparatus as recited in Claim 28, wherein said hash function includes a hash function h':$\{0, 1\}^* \rightarrow G \cup \{\perp\}$, that outputs an element of G or $\perp$ indicating a failure.

31. The apparatus as recited in Claim 24, wherein said signature generating logic is further configured to output said digital signature.

32. The apparatus as recited in Claim 31, further comprising:

signature verifying logic operatively coupled to receive said output digital signature and determine if said digital signature is valid.

33. The apparatus as recited in Claim 32, wherein said signature verifying logic is configured using said parameter data and said parameter data establishes a base group G and a generator g as system parameters for said signature verifying logic.

34. The apparatus as recited in Claim 33, wherein:

said public key data includes public key data v;

said identified data includes a message m;

said digital signature includes signature σ; and

said signature verifying logic determines if said digital signature is valid by determining $h \leftarrow h(m)$ using at least one hash function, and verifying that (g, v, h, σ) is a valid Gap Diffie-Hellman tuple.

35. The apparatus as recited in Claim 24, wherein said digital signature is included in a product ID.

36. A method comprising:

receiving message data and a corresponding digital signature and public key data;

using parameter data configure signature verifying logic that performs cryptography operations based on a Jacobian of a curve, said Jacobian having a genus greater than one, said parameter data causing said signature verifying logic to select at least one Gap Diffie-Hellman (GDH) group of elements relating to said curve; and

with said signature verifying logic, determining if said digital signature is valid using said public key data and said message data.

37. The method as recited in Claim 36, wherein said message data includes a message $m \in \{0, 1\}^*$.

38. The method as recited in Claim 37, wherein said parameter data establishes a base group G and a generator g as system parameters for said signature verifying logic.

39. The method as recited in Claim 38, wherein:

said public key data includes public key data v;

said digital signature includes signature σ; and

determining if said digital signature is valid further includes:

determining $h \leftarrow h(m)$ using at least one hash function, and

verifying that (g, v, h, σ) is a valid Gap Diffie-Hellman tuple.

40. The method as recited in Claim 39, wherein said hash function includes a full-domain hash function $h: \{0, 1\}^* \rightarrow G$.

41. The method as recited in Claim 39, wherein said hash function includes a hash function $h': \{0, 1\}^* \rightarrow G \cup \{\perp\}$, that outputs an element of G or $\perp$ indicating a failure.

42. A computer-readable medium having computer implementable instructions for causing at least one processing unit to perform acts comprising:

receiving message data and a corresponding digital signature and public key data;

using parameter data configure signature verifying logic that performs cryptography operations based on a Jacobian of a curve, said Jacobian having a genus greater than one, said parameter data causing said signature verifying logic to select at least one Gap Diffie-Hellman (GDH) group of elements relating to said curve; and

with said signature verifying logic, determining if said digital signature is valid using said public key data and said message data.

43. The computer-readable medium as recited in Claim 42, wherein said message data includes a message $m \in \{0, 1\}^*$.

44. The computer-readable medium as recited in Claim 43, wherein said parameter data establishes a base group G and a generator g as system parameters for said signature verifying logic.

45. The computer-readable medium as recited in Claim 44, wherein:

said public key data includes public key data v;

said digital signature includes signature σ; and

determining if said digital signature is valid further includes:

determining $h \leftarrow h(m)$ using at least one hash function, and

verifying that (g, v, h, σ) is a valid Gap Diffie-Hellman tuple.

46. The computer-readable medium as recited in Claim 45, wherein said hash function includes a full-domain hash function h: $\{0, 1\}^* \rightarrow G$.

47. The computer-readable medium as recited in Claim 45, wherein said hash function includes a hash function h':$\{0, 1\}^* \rightarrow G \cup \{\perp\}$, that outputs an element of G or $\perp$ indicating a failure.

48. A method comprising:

identifying data to be signed;

establishing parameter data for use with signature generating logic that encrypts data based on a Weil pairing on a Jacobian of at least one super-singular curve having a genus greater than one;

determining private key data and corresponding public key data using said signature generating logic; and

signing said identified data with said private key data using said signature generating logic to create a corresponding digital signature.

49. The method as recited in Claim 48, wherein said identified data includes a message $m \in \{0, 1\}^*$.

50. The method as recited in Claim 49, wherein said signature generating logic establishes E/F_p^l as an algebraic curve having genus g equal to at least two, J being a corresponding Jacobian, such that $P, Q \in J$ are linearly independent points of order q and $P \in J/F_{p^l}$ and $Q \in J/F_{p^{l\alpha}}$.

51. The method as recited in Claim 50, wherein determining said private key data and said public key data includes:

picking $x \in_R Z_q^*$, and

computing $R \leftarrow xQ$, wherein said public key data includes R and said private key data includes x.

52. The method as recited in Claim 51, wherein signing said identified data using with said private key data using said signature generating logic further includes:

determining $P_m \leftarrow h(m) \in J/F_{p'}$, and $S_m \leftarrow xP_m$, wherein said digital signature includes σ, which is an x-coordinate of g points in a representation of S_m as a reduced divisor.

53. The method as recited in Claim 48, further comprising:

outputting said digital signature.

54. The method as recited in Claim 53, further comprising:

determining if said digital signature is valid using signature verifying logic.

55. The method as recited in Claim 54, wherein said signature verifying logic is configured to:

receive said public key as R, said identified data as a message m, and said digital signature as σ;

determine that said digital signature is valid for message m using said public key data R, if $u = v$ after letting S be a point on J/F_{p^l} whose x-coordinates is in σ and whose y-coordinate is y for some $y \in F_{p^l}$, and by setting $u \leftarrow e(P,S)$ and $v \leftarrow e(R, \phi(h(m)))$;

otherwise determining that said digital signature σ is invalid.

56. The method as recited in Claim 48, wherein said digital signature is included in a product ID.

57. A computer-readable medium having computer implementable instructions for causing at least one processing unit to perform acts comprising:

identifying data to be signed;

establishing parameter data for use with signature generating logic that encrypts data based on a Weil pairing on a Jacobian of at least one super-singular curve having a genus greater than one;

determining private key data and corresponding public key data using said signature generating logic; and

signing said identified data with said private key data using said signature generating logic to create a corresponding digital signature.

58. The computer-readable medium as recited in Claim 57, wherein said identified data includes a message m $\in$ {0, 1}*.

59. The computer-readable medium as recited in Claim 58, wherein said signature generating logic establishes E/F_p^l as an algebraic curve having genus g

equal to at least two, J being a corresponding Jacobian, such that $P, Q \in J$ are linearly independent points of order q and $P \in J/F_{p^l}$ and $Q \in J/F_{p^{la}}$.

60. The computer-readable medium as recited in Claim 59, wherein determining said private key data and said public key data includes:

picking $x \xleftarrow{R} Z_q^*$, and

computing $R \leftarrow xQ$, wherein said public key data includes R and said private key data includes x.

61. The computer-readable medium as recited in Claim 60, wherein signing said identified data using with said private key data using said signature generating logic further includes:

determining $P_m \leftarrow h(m) \in J/F_{p^l}$, and $S_m \leftarrow xP_m$, wherein said digital signature includes σ, which is an x-coordinate of g points in a representation of S_m as a reduced divisor.

62. The computer-readable medium as recited in Claim 57, further comprising:

outputting said digital signature.

63. The computer-readable medium as recited in Claim 62, further comprising:

determining if said digital signature is valid using signature verifying logic.

64. The computer-readable medium as recited in Claim 63, wherein said signature verifying logic is configured to:

receive said public key as R, said identified data as a message m, and said digital signature as σ;

determine that said digital signature is valid for message m using said public key data R, if $u = v$ after letting S be a point on J/F_{p^l} whose x-coordinates is in σ and whose y-coordinate is y for some $y \in F_{p^l}$, and by setting $u \leftarrow e(P,S)$ and $v \leftarrow e(R, \phi(h(m)))$;

otherwise determining that said digital signature σ is invalid.

65. An apparatus comprising:

memory configured to store identifying data to be signed;

signature generating logic that is configured using parameter data such that said signature generating logic encrypts data based on a Weil pairing on a Jacobian of at least one super-singular curve having a genus greater than one, and determines private key data and corresponding public key data and signs said identified data with said private key data using said signature generating logic to create a corresponding digital signature.

66. The apparatus as recited in Claim 65, wherein said identified data includes a message $m \in \{0, 1\}^*$.

67. The apparatus as recited in Claim 66, wherein said signature generating logic establishes E/F_p^l as an algebraic curve having genus g equal to at

least two, J being a corresponding Jacobian, such that $P, Q \in J$ are linearly independent points of order q and $P \in J/F_{p^l}$ and $Q \in J/F_{p^{l\alpha}}$.

68. The apparatus as recited in Claim 67, wherein said signature generating logic is further configured to:

pick $x \overset{R}{\leftarrow} Z_q^*$, and

determine $R \leftarrow xQ$, wherein said public key data includes R and said private key data includes x.

69. The apparatus as recited in Claim 68, wherein said signature generating logic is further configured to:

determine $P_m \leftarrow h(m) \in J/F_{p^l}$, and $S_m \leftarrow xP_m$, wherein said digital signature includes σ, which is an x-coordinate of g points in a representation of S_m as a reduced divisor.

70. The apparatus as recited in Claim 65, wherein said signature generating logic is further configured to:

output said digital signature.

71. The apparatus as recited in Claim 70, further comprising:

signature verifying logic configured to receive said output digital signature and determine if said digital signature is valid.

72. The apparatus as recited in Claim 71, wherein said signature verifying logic is configured to:

receive said public key as R, said identified data as a message m, and said digital signature as σ;

determine that said digital signature is valid for message m using said public key data R, if $u = v$ after letting S be a point on J/F_{p^l} whose x-coordinates is in σ and whose y-coordinate is y for some $y \in F_{p^l}$, and by setting $u \leftarrow e(P,S)$ and $v \leftarrow e(R, \phi(h(m)))$;

otherwise determining that said digital signature σ is invalid.

73. The apparatus as recited in Claim 65, wherein said digital signature is included in a product ID.